

GT 5.0.0 SimpleCA: Admin Guide

GT 5.0.0 SimpleCA: Admin Guide

Introduction

This guide contains advanced configuration information for system administrators working with SimpleCA. It provides references to information on procedures typically performed by system administrators, including installation, configuring, deploying, and testing the installation.



Important

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [Installing GT 5.0.0](#). Read through this guide before continuing!

The following are instructions for how to use SimpleCA to request and sign *certificate* for a GT 5.0.0 installation.

SimpleCA provides an easy way to create and package a *Certificate Authority (CA)*, as well as tools for requesting and signing user and host certificates. It is similar to OpenSSL's **CA.sh** command but includes support for packaging the CA certificate, signing policy file, and configuration information needed by clients to request certificates. You can find other CA options in [Obtaining host certificates](#).

Table of Contents

1. Building and Installing	1
1. Create users	1
2. Run the setup script	1
3. Host certificates	4
4. User certificates	4
2. Configuring	7
1. Configure SimpleCA for multiple machines	7
3. Testing	8
I. Simple CA Commands	9
grid-ca-sign	10
4. Security Considerations	11
1. Security considerations for SimpleCA	11
Glossary	12

List of Tables

1.1. CA Name components 1

Chapter 1. Building and Installing

1. Create users

Make sure you have the following users on your machine:

- Your *user* account, which will be used to run the client programs.
- A generic *globus* account, which will be used to perform administrative tasks. This user will also be in charge of managing the SimpleCA. To do this, make sure this account has read and write permissions in the `$GLOBUS_LOCATION` directory.

2. Run the setup script

A script was installed to set up a new SimpleCA. You only need to run this script *once* to create a certificate authority that can be used on multiple computers or grids.

Run the setup script:

```
$GLOBUS_LOCATION/setup/globus/setup-simple-ca
```

2.1. Configure the subject name

This script prompts you for information about the CA you wish to create:

```
The unique subject name for this CA is:
cn=Globus Simple CA, ou=simpleCA-grid.example.org, ou=GlobusTest, o=Grid
```

```
Do you want to keep this as the CA subject (y/n) [y]:
```

where:

Table 1.1. CA Name components

cn	Represents "common name". It identifies this particular certificate as the <i>CA Certificate</i> within the "GlobusTest/simpleCA-grid.example.org" domain, which in this case is Globus Simple CA.
ou	Represents "organizational unit". It identifies this CA from other CAs created by SimpleCA by other people. The second "ou" is specific to your hostname (in this case GlobusTest).
o	Represents "organization". It identifies the Grid.

Press **y** to keep the default subject name (recommended).

2.2. Configure the CA's email

The next prompt looks like:

Enter the email of the CA (this is the email where certificate requests will be sent to be signed by the CA):

Enter the email address where you intend to receive certificate requests. It should be your real email address that you check, not the address of the globus user.

2.3. Configure the expiration date

Then you'll see:

The CA certificate has an expiration date. Keep in mind that once the CA certificate has expired, all the certificates signed by that CA become invalid. A CA should regenerate the CA certificate and start re-issuing ca-setup packages before the actual CA certificate expires. This can be done by re-running this setup script. Enter the number of DAYS the CA certificate should last before it expires. [default: 5 years (1825 days)]:

This is the number of days for which the CA certificate is valid. Once this time expires, the CA certificate will have to be recreated and all of its certificates regranted.

Accept the default (recommended).

2.4. Enter a passphrase

Next you'll see:

```
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to '/home/globus/.globus/simpleCA/private/cakey.pem'
Enter PEM pass phrase:
```

The passphrase of the CA certificate will be used only when signing certificates (with **grid-cert-sign**). It should be hard to guess, as its compromise will compromise all the certificates signed by the CA.

Enter your passphrase.



Important:

Your passphrase must *not* contain any spaces.

2.5. Confirm generated certificate

Finally you'll see the following:

A self-signed certificate has been generated for the Certificate Authority with the subject:

```
/O=Grid/OU=GlobusTest/OU=simpleCA-grid.example.org/CN=Globus Simple CA
```

If this is invalid, rerun this script

```
setup/globus/setup-simple-ca
```

and enter the appropriate fields.

The private key of the CA is stored in /home/globus/.globus/simpleCA//private/cakey.pem
The public CA certificate is stored in /home/globus/.globus/simpleCA//cacert.pem

The distribution package built for this CA is stored in

```
/home/globus/.globus/simpleCA//globus_simple_ca_68ea3306_setup-0.17.tar.gz
```

This information will be important for setting up other machines in your grid. The number *68ea3306* in the last line is known as your *CA hash*. It will be an 8 hexadecimal digit string.

Press any key to acknowledge this screen.

Your CA setup package finishes installing and ends the procedure with the following reminder:

Note: To complete setup of the GSI software you need to run the following script as root to configure your security configuration directory:

```
/opt/gt4/setup/globus_simple_ca_68ea3306_setup/setup-gsi
```

For further information on using the setup-gsi script, use the -help option. The -default option sets this security configuration to be the default, and -nonroot can be used on systems where root access is not available.

```
setup-ssl-utils: Complete
```

We'll run the setup-gsi script in the next section. For now, just notice that it refers to your `$GLOBUS_LOCATION` and the *CA Hash* from the last message.

2.6. Complete setup of GSI

To finish the setup of GSI, we'll run the script noted in the previous step.

Run the following as root (or, if no root privileges are available, add the **-nonroot** option to the command line):

```
$GLOBUS_LOCATION/setup/globus_simple_ca_CA_Hash_setup/setup-gsi -default
```

The output should look like:

```
setup-gsi: Configuring GSI security
Installing /etc/grid-security/certificates//grid-security.conf.CA_Hash...
Running grid-security-config...
Installing Globus CA certificate into trusted CA certificate directory...
Installing Globus CA signing policy into trusted CA certificate directory...
setup-gsi: Complete
```

3. Host certificates

You must request and sign a *host certificate* and then copy it into the appropriate directory for secure services. The certificate must be for a machine which has a consistent name in DNS; you should not run it on a computer using DHCP, where a different name could be assigned to your computer.

3.1. 3.1 Request a host certificate

As root, run:

```
grid-cert-request -host 'hostname'
```

This creates the following files:

- /etc/grid-security/hostkey.pem
- /etc/grid-security/hostcert_request.pem
- (an empty) /etc/grid-security/hostcert.pem

Note: If you are using your own CA, follow their instructions about creating a hostcert (one which has a commonName (CN) of your hostname), then place the cert and key in the /etc/grid-security/ location. You may then proceed to [Section 4, “User certificates”](#).

3.2. Sign the host certificate

1. As globus, run:

```
grid-ca-sign -in /etc/grid-security/hostcert_request.pem -out hostsigned.pem
```

2. A signed host certificate, named `hostsigned.pem`, is written to the current directory.
3. When prompted for a passphrase enter the one you specified in [Section 2.4, “Enter a passphrase”](#) (for the private key of the CA certificate).
4. As root move the signed host certificate to `/etc/grid-security/hostcert.pem`.

The certificate should be owned by root and be read-only for other users.

The key should be read-only by root.

4. User certificates

Users also must request *user certificates*, which you will sign using the *globus* user.

4.1. Request a user certificate

As your normal user account (*not globus*), run:

```
% grid-cert-request
```

A private key and a certificate request has been generated with the subject:

```
/O=Grid/OU=GlobusTest/OU=simpleCA-grid.example.org/CN=Joe User
```

The private key is stored in /home/juser/.globus/hostkey.pem
The request is stored in /home/juser/.globus/hostcert_request.pem

Please e-mail the request to the Example CA `grid@example.org`
You may use a command similar to the following:

```
cat /tmp/example/hostcert_request.pem | mail grid@example.org
```

Only use the above if this machine can send AND receive e-mail. if not, please mail using some other method.

Your certificate will be mailed to you within two working days.
If you receive no response, contact Example CA at `grid@example.org`

After you enter a passphrase, this creates

- `$HOME/.globus/usercert.pem` (empty)
- `$HOME/.globus/userkey.pem`
- `$HOME/.globus/usercert_request.pem`

Email the `usercert_request.pem` file to the CA administrator maintainer.

4.2. Sign the user certificate

1. As the SimpleCA owner *globus*, run:

```
% grid-ca-sign -in usercert_request.pem -out signed.pem
```

2. When prompted for a password enter the one you specified in [Section 2.4, “Enter a passphrase”](#) (for the private key of the CA certificate).
3. Now send the signed copy (`signed.pem`) back to the user who requested the certificate.
4. As your normal user account (*not globus*), copy the signed user certificate into `>HOME/.globus/` and rename it as `usercert.pem`, thus replacing the empty file.

The certificate should be owned by the user and be read-only for other users.

The key must be read-only by the owner.

Chapter 2. Configuring

1. Configure SimpleCA for multiple machines

So far, you have a single machine configured with SimpleCA certificates. Recall that in [Section 2.5, “Confirm generated certificate”](#) a CA setup package was created in `.globus/simpleCA/globus_simple_ca_HASH_setup-0.17.tar.gz`. If you want to use your certificates on another machine, you must install that CA setup package on that machine.

To install it, copy that package to the second machine and run:

```
$GLOBUS_LOCATION/sbin/gpt-build globus_simple_ca_HASH_setup-0.17.tar.gz gcc32dbg  
$GLOBUS_LOCATION/sbin/gpt-postinstall
```

Then you will have to perform **setup-gsi -default** from [Section 2.6, “Complete setup of GSI”](#).

If you are going to run services on the second host, it will need its own host certificate ([Section 3, “Host certificates”](#)) and grid-mapfile (as described in the basic configuration instructions in [Section 4, “Add authorization”](#)).

You may re-use your *user certificates* on the new host. You will need to copy the requests to the host where the SimpleCA was first installed in order to sign them.

Chapter 3. Testing

To verify that the SimpleCA certificate is installed in `/etc/grid-security/certificates` and that your certificate is in place with the correct permissions, run:

```
user$ grid-proxy-init -debug -verify
```

After entering your passphrase, successful output looks like:

```
% grid-proxy-init -debug -verify
```

```
User Cert File: /home/juser/.globus/usercert.pem
```

```
User Key File: /home/juser/.globus/userkey.pem
```

```
Trusted CA Cert Dir: /etc/grid-security/certificates
```

```
Output File: /tmp/x509up_u1817
```

```
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-grid.example.org/CN=Joe User
```

```
Enter GRID pass phrase for this identity:
```

```
Creating proxy .....+++++
```

```
.....+++++
```

```
Done
```

```
Proxy Verify OK
```

```
Your proxy is valid until: Sat Mar 20 03:01:46 2009
```

Simple CA Commands

Name

grid-ca-sign -- Sign a certificate

grid-ca-sign [-help] [-force] [-dir *CA-DIR*] [-openssl-help] [*OPENSSL-OPTION...*] -in *CERTIFICATE-REQUEST* -out *CERTIFICATE*

Description

The **grid-ca-sign** program creates and signs a *certificate* from a certificate request. There are two required command-line parameters: **-in** *CERTIFICATE-REQUEST* the name of a file containing the X.509 Certificate request, and **-out** *CERTIFICATE* the name of the file to write the signed certificate to. By default, the new certificate is signed by the *CA Certificate* located in `$HOME/.globus/simpleCA`.

grid-ca-sign supports these options:

-help	Display help and then exit
-force	Overwrite <i>CERTIFICATE</i> if it already exists.
-dir <i>CA-DIR</i>	Use the CA certificate in <i>CA-DIR</i> to sign the certificate request
-openssl-help	Display the command-line help to the openssl program. grid-ca-sign will pass unrecognized command-line options to openssl .
-in <i>CERTIFICATE-REQUEST</i>	Read and sign the certificate request in the file <i>CERTIFICATE-REQUEST</i> .
-out <i>CERTIFICATE</i>	Write the signed certificate to <i>CERTIFICATE</i> .

Examples

Sign a certificate request in the file `req001.pem` and write the certificate to `cert001.pem`:

```
% grid-ca-sign -in req001.pem -out cert001.pem
```

To sign the request
please enter the password for the CA key:

The new signed certificate is at: `/home/gridCA/.globus/simpleCA/newcerts/01.pem`

```
%
```

Limitations

Not all OpenSSL options will work, as some are used internally by grid-ca-sign.

Chapter 4. Security Considerations

1. Security considerations for SimpleCA

The operator of a CA must protect the private key of the CA. It should not be stored unencrypted or on a network filesystem.

Simple CA enforces the subject name policies in the simple CA's configuration files. If modified, the signing_policy file distributed to clients of the CA must also be modified.

Glossary

C

Certificate Authority (CA)	An entity that issues certificates.
CA Certificate	The CA's certificate. This certificate is used to verify signature on certificates issued by the CA. GSI typically stores a given CA certificate in <code>/etc/grid-security/certificates/<hash>.0</code> , where <code><hash></code> is the hash code of the CA identity.
certificate	A public key plus information about the certificate owner bound together by the digital signature of a CA. In the case of a CA certificate, the certificate is self signed, i.e. it was signed using its own private key.

P

private key	The private part of a key pair. Depending on the type of certificate the key corresponds to it may typically be found in <code>\$HOME/.globus/userkey.pem</code> (for user certificates), <code>/etc/grid-security/hostkey.pem</code> (for host certificates) or <code>/etc/grid-security/<service>/<service>key.pem</code> (for service certificates).
-------------	---

For more information on possible private key locations see [this](#).

U

user certificate	A EEC belonging to a user. When using GSI, this certificate is typically stored in <code>\$HOME/.globus/usercert.pem</code> . For more information on possible user certificate locations, see this .
------------------	---